

# Assessing GitLab-Jira Integration

This document compares the three possibilities we have regarding issue tracking and management, including an assessment of the viability of [integrating GitLab with Jira](#).

The table below summarizes the assessment. Mainly, there are two things being evaluated: 1. How GitLab-only issue tracking compares to having GitLab and Jira and 2. in the case of the latter, how having GitLab separate from Jira compares to having it integrated with Jira.

**MR:** Merge Request (equivalent of the hub's Pull Request).

	GitLab-only	Separate GitLab and Jira	Integrated GitLab and Jira
Pros:	<ul style="list-style-type: none"><li>• One login step.</li><li>• One user interface /website to get used to (not counting with internal company trackers).</li><li>• Complete and unified view of both code and tasks (issues, features, bugs, filtering, code reviews, etc.).</li><li>• Most resources are linkable.</li><li>• Both code and tasks can be associated to labels, milestones and releases.</li><li>• End-to-end: from issue creation and triaging all the way to generating a release triggered by CD.</li><li>• Less clutter as it's a single tool.</li></ul>	<ul style="list-style-type: none"><li>• One login step, for project management only.</li><li>• One user interface /website to get used to, for project management only (not counting with internal company trackers).</li><li>• Hierarchical issues support.</li></ul>	<ul style="list-style-type: none"><li>• One login step, for project management only.</li><li>• One user interface/website to get used to, for project management only (not counting with internal company trackers).</li><li>• Hierarchical issues support.</li><li>• Linking to and from Jira/GitLab will, most of the time, be 1 click away.</li><li>• Jira issues can also be viewed inside GitLab as read-only*.</li><li>• Developers can write the name and number of the Jira issue in many different places (commit messages, MR text, comment, etc.) to automatically create a hyperlink to the respective Jira issue.</li><li>• Jira issues will receive updates in the form of Jira comments for certain trigger actions in GitLab such as a new commit being pushed or merged**.</li><li>• Can be configured to automatically update the status of a Jira issue when a certain GitLab event, such as closing an issue when an MR gets merged, although such an automated action may backfire by updating issues prematurely**.</li></ul>
Cons:	<ul style="list-style-type: none"><li>• It's not the industry-proven Jira, which comes with unspecified risk.</li><li>• Doesn't (<a href="#">yet.. there's work ongoing..</a>) explicitly support hierarchical issues (think epics vs. stories) although through labeling and linking, the same end-result can be achieved.</li></ul>	<ul style="list-style-type: none"><li>• Two login steps for developers (GitLab, then Jira).</li><li>• Two user interface /website to get used to, for developers.</li><li>• Hard (or in some cases impossible) to go to/from GitLab and Jira from one another (easier to just click a browser bookmark).</li><li>• Confusion in user interface as Issues list will seem to be abandoned or simply not visible, at first glance.</li><li>• Not welcoming to technical new contributors and bug reporters (i.e. quickly report an issue and immediately submit a fix for the same).</li></ul>	<ul style="list-style-type: none"><li>• Two login steps for developers (GitLab, then Jira).</li><li>• Two user interface/websites to get used to, for developers.<ul style="list-style-type: none"><li>◦ However, being 1 click away from switching is helpful.</li></ul></li><li>• Even though the GitLab Issues list can be hidden, certain UI elements which are tightly-coupled to MRs will remain. For example, MRs will still expect to be associated to labels, milestones and releases. They are useful for querying and filtering MRs and Issues. As such, even the Integrated Jira will still leave contributors slightly confused or force them to take awkward steps. This is in contrast to what they would expect from a pure GitLab project, where task management, code contribution and code review are unified and 1 click away.<ul style="list-style-type: none"><li>◦ Actual example of awkward steps from above: Visibility - with Jira, we can't filter by milestones or labels when navigating merge requests. That info is in Jira. And we can't see what labels or milestone are assigned to a particular MR, unless we visit the respective Jira.</li></ul></li></ul>

\*In order to be able to see Jira issues inside GitLab, the GitLab Premium subscription is required (EMCO currently uses the Free tier)

\*\*Currently, we can't truly integrate GitLab with Jira unless access API access to the LFN Jira can be granted.

Milestone List view in GitLab to illustrate one of the ways GitLab unifies releases, tasks and deliverables:

Open 1 Closed 0 All 1

Filter by milestone name

Due soon



New milestone

21.12

expires on Dec 15, 2021

Igor D.C. / emco0000



33% complete

Close Milestone